Appendix I. Data Compression Formats

This appendix provides a brief description of each of the compression formats that has been approved by the PDS for archive data.

Each section in this appendix includes a high level description of the compression format, PDS-specific implementation rules, and information about how to properly label files implementing the compression algorithm. Each section should also include a sample label.

Chapter Contents

I.1 CLEM-JPEG

TBD

I.1.1 PDS Implementation Rules

TBD

I.1.2 Labeling

TBD

I.1.3 Label Example

TBD

I.2 HUFFMAN FIRST DIFFERENCE

TBD

I.2.1 PDS Implementation Rules

TBD

I.2.2 Labeling

TBD

I.2.3 Label Example

TBD

I.3 JPEG 2000

JPEG 2000 is defined as an "image coding system". The ISO/IEC specification describing it includes not only the syntax for a compressed image codestream (mime type "J2C"), but also a description of the binary "JP2" file format that may be used to enhance the utility of the codestream.

Unlike many older compression algorithms, JPEG 2000 provides a great deal of flexibility in the way in which data may be stored in the codestream and retrieved from it. This flexibility allows for the progressive decompression of "layers" of the image with increasing resolution or precision. It also permits the extraction and decompression of only a portion or "tile" of the image. Specific portions of the image of particular interest to the intended audience may also be stored at the beginning of the codestream so that they may be accessed and decompressed first. (This would be of potential interest for approach images where the target of the observation fills only a small portion of the field of view.)

All of the information necessary to successfully decompress a JPEG 2000 image is contained in the J2C codestream. However, the information necessary to take advantage of the additional capabilities is only available in the JP2 format.

A JP2 file essentially consists of a set of "boxes" that encapsulate both the J2C codestream and the meta data that describe it (Figure I.1). The first two of these boxes provide information that identifies the file as a JP2 formatted file. The following "superbox" is the JP2 header box which contains information about the image size, resolution, colorspace, etc. Following this, in no particular order, are contiguous codestream boxes containing the compressed image data and (optionally) intellectual property rights boxes, XML boxes containing vendor-specific meta data, and UUID boxes containing reference URLs. In this document, all of these non-image boxes will be collectively referred to as the "JP2 binary wrapper."

PDS requires the presence of the JP2 binary wrapper so that external software may take full advantage of the JPEG 2000 capabilities. PDS software will have the capability to fully decompress the entire data file, but will not necessarily have the capability to decompress subsets of the codestream such as individual resolution layers or tiles.

The ISO/IEC specification defining JPEG 2000 is entitled "Information technology – JPEG 2000 image coding system" and may be ordered from the ISO by going to their web site (http://www.iso.ch/) and searching on "JPEG 2000".

I.3.1 Table of Compression Ratios

The JPEG 2000 compression algorithm was tested on two Mars Express HRSC images. In both cases, the binary headers and line prefix information on these images were retained in order to provide some additional stress testing of the compression algorithm. With the inclusion of this artificially included binary noise, both of the following images cover the full 16-bit data range.

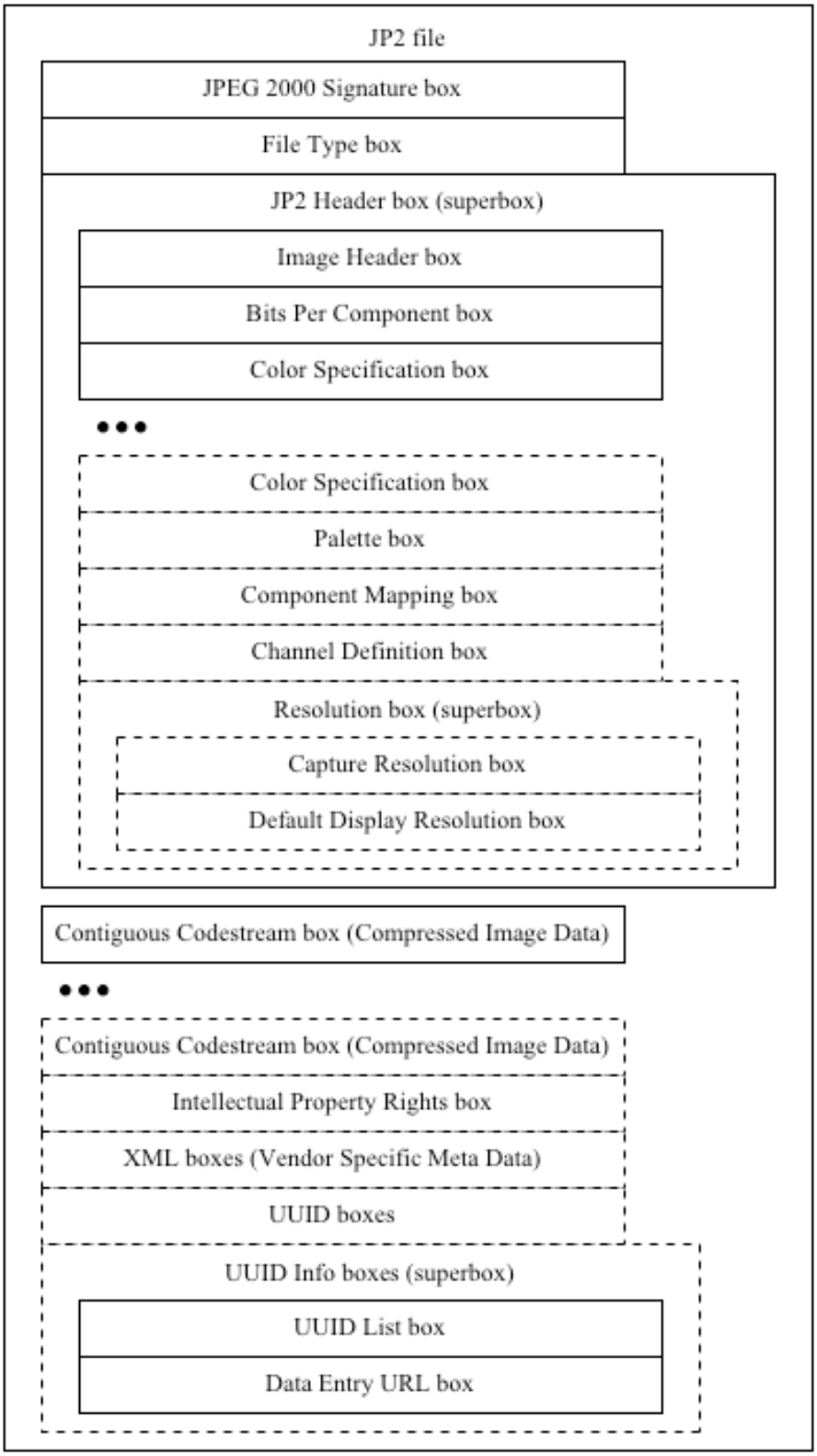

Figure I.1 – Graphical representation of a JP2 file. Dashed lines indicate optional boxes. (Modified from ISO/IEC 15444-1:2004, "Information technology – JPEG 2000 image coding system: Core coding system", figure T.800_FI.1)

The first image, “h0068_0000_s22.img”, has 2,618 samples and 119,757 lines, with 16 bit pixels. The image data alone have a dynamic range of 67 to 308 DN, with a standard deviation of 52.5.

The second image, “h0068_0009_s22.img”, has 2,618 samples and 11,013 lines, with 16 bit pixels. The image data alone have a dynamic range of 62 to 188 DN, with a standard deviation of 9.2

Both images were also converted to 8 bit data for comparison purposes. Selecting for various tile sizes in the JPEG 2000 compression software, the following lossless compression ratios were obtained:

	h0068_0000_s22		**h0068_0009_s22**	
tile size	16-bit	8-bit	16-bit	8-bit
1024	5.83	2.89	6.00	2.25
512	5.82	2.88	5.99	2.25
256	5.78	2.87	5.94	2.24
128	5.64	2.81	5.80	2.20

These results can be compared with a compression ratio of 3.8 for both of the 16-bit versions of the h0068_0000_s22 and h0068_0009_s22 images, when using Zip compression.

Re-projected images containing large areas of no data and half word image data which do not utilize a full 16-bit data range should result in even higher compression ratios.

I.3.2 PDS Implementation Rules

The JPEG 2000 compression algorithm must be implemented on PDS archive volumes as a codestream encapsulated by the JP2 binary wrapper (ie., not as a bare codestream).

Only lossless compression may be used.

Furthermore, the syntax and features of the compressed codestream must conform to part 1, the “Core coding system”, of the ISO/IEC specification defining JPEG 2000, namely 15444-1.

Use of the JPEG 2000 compression algorithm and format is restricted to derived image data where the source, or EDR, products have been archived in an uncompressed format.

I.3.3 Labeling

For each image archived in JPEG 2000 format, two files need to be considered: (1) the compressed file physically included in the archive, and (2) the dynamically generated data file produced by decompressing the JP2 file. These two files have the same name but different extensions: “.JP2” for a JP2 formatted file and “IMG” for the decompressed file. (The “JP2” file

extension is reserved exclusively, and must be used, for JPEG 2000 compressed files within the PDS).

Like all PDS data files, both the compressed and the decompressed data files require labels. Both files must be described by a single, detached PDS label file using the combined-detached label approach (see Section 5.2.2). Attached labels are not permitted for JPEG 2000 compressed data, because an attached PDS header would violate the JP2 format. In a combined-detached label, each individual file is described using an explicit FILE object. The general framework is:

```
PDS_VERSION_ID                = PDS3
DATA_SET_ID                   = ...
PRODUCT_ID                    = ...
  (other parameters relevant to both compressed and decompressed files)

OBJECT                        = COMPRESSED_FILE
  (parameters describing the compressed file)
END_OBJECT                    = COMPRESSED_FILE

OBJECT                        = UNCOMPRESSED_FILE
  (parameters describing the uncompressed file)
END_OBJECT                    = UNCOMPRESSED_FILE
END
```

The compressed file is described by a “minimal label” (see Section 5.2.3), and the following keywords are required:

```
FILE_NAME                     = name of the compressed file
RECORD_TYPE                   = UNDEFINED
ENCODING_TYPE                 = "JP2"
ENCODING_TYPE_VERSION_NAME    = version of the JPEG 2000 specification
                                consistent with the data product
INTERCHANGE_FORMAT            = BINARY
UNCOMPRESSED_FILE_NAME        = name of the decompressed file
REQUIRED_STORAGE_BYTES        = approximate total number of bytes in the
                                decompressed data file
DESCRIPTION                   = brief description of the JPEG 2000
                                format, including a reference to the
                                full specification
```

Typically, the DESCRIPTION is given as a pointer to a file called “JP2INFO.TXT” found in the DOCUMENT directory on the same volume.

The subsequent UNCOMPRESSED_FILE object contains a complete description of the data file obtained by decompressing the JPEG 2000 file.

I.3.4 Label Example

The following combined detached label describes a hypothetical JP2 formatted image and the decompressed PDS formatted image derived from it:

```
PDS_VERSION_ID                = PDS3
```

```
/* IDENTIFICATION DATA ELEMENTS */

MISSION_NAME                  = "MARS RECONNAISSANCE ORBITER"
INSTRUMENT_HOST_NAME          = "MARS RECONNAISSANCE ORBITER"
INSTRUMENT_NAME               = "HIGH RESOLUTION IMAGING SCIENCE
                                 EXPERIMENT"
TARGET_NAME                   = "MOON"
DATA_SET_ID                   = "MRO-L-HIRISE-5-DIM-V1.0"
PRODUCT_ID                    = "CRU_000004_1200_RED2_2"
START_TIME                    = 2005-09-08T23:16:44.863
STOP_TIME                     = 2005-09-08T23:16:51.569
SPACECRAFT_CLOCK_START_COUNT  = 810688604:56542
SPACECRAFT_CLOCK_STOP_COUNT   = 810688611:37300
PRODUCT_CREATION_TIME         = 2005-09-09T15:35:45
(etc.)

/* DESCRIPTIVE DATA ELEMENTS */

(etc.)


OBJECT                        = COMPRESSED_FILE
  FILE_NAME                   = "FILENAME.JP2"
  RECORD_TYPE                 = UNDEFINED
  ENCODING_TYPE               = "JP2"
  ENCODING_TYPE_VERSION_NAME  = "ISO/IEC15444-1:2004"
  INTERCHANGE_FORMAT          = BINARY
  UNCOMPRESSED_FILE_NAME      = "FILENAME.IMG"
  REQUIRED_STORAGE_BYTES      = 2400000000
  ^DESCRIPTION                = "JP2INFO.TXT"
END_OBJECT                    = COMPRESSED_FILE

OBJECT                        = UNCOMPRESSED_FILE
  FILE_NAME                   = "FILENAME.IMG"
  RECORD_TYPE                 = FIXED_LENGTH
  RECORD_BYTES                = 40000
  FILE_RECORDS                = 60000

/* POINTERS TO DATA OBJECTS */

  ^IMAGE                      = "FILENAME.IMG"

/* DATA OBJECT DEFINITIONS */

  OBJECT                      = IMAGE
    LINES                     = 60000
    LINE_SAMPLES              = 20000
    SAMPLE_TYPE               = UNSIGNED_INTEGER
    SAMPLE_BITS               = 16
    SAMPLE_BIT_MASK           = 2#0011111111111111#
    (etc.)
  END_OBJECT                  = IMAGE
END_OBJECT                    = UNCOMPRESSED_FILE
END
```

I.4 PREVIOUS PIXEL

TBD

I.4.1 PDS Implementation Rules

TBD

I.4.2 Labeling

TBD

I.4.3 Label Example

TBD

I.5 RUN LENGTH

TBD

I.5.1 PDS Implementation Rules

TBD

I.5.2 Labeling

TBD

I.5.3 Label Example

TBD

I.6 ZIP

The Zip method was chosen because the algorithm and supporting software for all major platforms are available without charge to the general user community. The *Info-Zip Consortium* and Info-Zip working group, for example, provide information and software at this URL:

http://www.info-zip.org

This same information is available on line from PDS at:

http://pds.jpl.nasa.gov

I.6.1 PDS Implementation Rules

A volume containing zip files with combined-detached labels as presented below conforms to all established PDS standards *provided both the zip file and its constituent data files are archived*. The unique feature of a Zip-compressed PDS archive volume is that only the zip files appear; the UNCOMPRESSED_FILE objects described by the labels are not present on the volume, but can be obtained by unzipping the zip files provided.

In the interests of long-term archiving, a PDS archive zip file must include all the support files required to completely reconstitute the labeled data files. Specifically, the zipped archive must include not only the data files, but also the label file(s) for the uncompressed data. Ideally, any .FMT files referenced by ^STRUCTURE keywords in the labels should also be included in the zip file.

Note: These additional .LBL and .FMT files do not need to be described by UNCOMPRESSED_FILE objects in the label, because PDS label and format files never require labels. Furthermore, the sizes of these files do not need to be included in the value of the REQUIRED_STORAGE_BYTES keyword. However, the names of these files do need to be included in the list of UNCOMPRESSED_FILE_NAME values.

I.6.2 Labeling

When archiving data in Zip format, two files need to be considered: (1) the zip file itself, and (2) the data file produced by decompressing the zip file. PDS strongly recommends that these two files have the same name but different extensions: ".ZIP" for the zip file and a more descriptive extension (e.g., ".DAT" or ".IMG") for the unzipped file. The ".ZIP" file extension is reserved exclusively for zip-compressed files within the PDS.

PDS does not recommend the practice of compressing multiple data files into a single zip file, unless those files reside in the same directory and have the same name, but different extensions.

For example, if file “ABC.IMG” contains an image and file “ABC.TAB” contains a table of additional information relevant to that image, then both files can be archived in the file “ABC.ZIP”. This will minimize the potential confusion for a user who may not be able to locate a desired file because it is hidden inside a zip file with a different name.

Like all PDS data files, both the zipped and the unzipped data files require labels. Both files must be described by a single, detached PDS label file using the combined-detached label approach (see Section 5.2.2). Attached labels are not permitted for Zip-compressed data, because the user must be able to examine the label before deciding whether or not to decompress the file. In a combined-detached label, each individual file is described as a FILE object. Here is the general framework:

```
PDS_VERSION_ID          = PDS3
DATA_SET_ID             = ...
PRODUCT_ID              = ...
   (other parameters relevant to both Zipped and Unzipped files)

OBJECT                  = COMPRESSED_FILE
   (parameters describing the compressed file)
END_OBJECT              = COMPRESSED_FILE

OBJECT                  = UNCOMPRESSED_FILE
   (parameters describing the first uncompressed file)
END_OBJECT              = UNCOMPRESSED_FILE

OBJECT                  = UNCOMPRESSED_FILE
   (parameters describing a second uncompressed file, if present)
END_OBJECT              = UNCOMPRESSED_FILE
END
```

The first FILE object, the COMPRESSED_FILE, refers to the zipped file; additional FILE objects, called UNCOMPRESSED_FILEs, refer to the decompressed data file(s) that the user will obtain by unzipping the first.

The zip file is described via a “minimal label” (see Section 5.2.3). The following keywords are required:

```
FILE_NAME               = name of the zipfile
RECORD_TYPE             = UNDEFINED
ENCODING_TYPE           = ZIP
INTERCHANGE_FORMAT      = BINARY
UNCOMPRESSED_FILE_NAME  = a list of the names of all the files archived
                          in the zipfile
REQUIRED_STORAGE_BYTES  = approximate total number of bytes in the data
                          files
DESCRIPTION             = a brief description of the zipfile format
```

Typically, the DESCRIPTION is given as a pointer to a file called “ZIPINFO.TXT” found in the DOCUMENT directory on the same volume.

The subsequent UNCOMPRESSED_FILE object(s) contain complete descriptions of the data files obtained by unzipping the zip file.

I.6.3 Label Example

The following is an example of a PDS label for a Zip-compressed data file.

```
PDS_VERSION_ID             = PDS3
DATA_SET_ID                = "HST-S-WFPC2-4-RPX-V1.0"
SOURCE_FILE_NAME           = "U2ON0101T.SHF"
PRODUCT_TYPE               = OBSERVATION_HEADER
PRODUCT_CREATION_TIME      = 1998-01-31T12:00:00

OBJECT                     = COMPRESSED_FILE
  FILE_NAME                = "0101_SHF.ZIP"
  RECORD_TYPE              = UNDEFINED
  ENCODING_TYPE            = ZIP
  INTERCHANGE_FORMAT       = BINARY
  UNCOMPRESSED_FILE_NAME   = {"0101_SHF.DAT", "0101_SHF.LBL"}
  REQUIRED_STORAGE_BYTES   = 34560
  ^DESCRIPTION             = "ZIPINFO.TXT"
END_OBJECT                 = COMPRESSED_FILE

OBJECT                     = UNCOMPRESSED_FILE
  FILE_NAME                = "0101_SHF.DAT"
  RECORD_TYPE              = FIXED_LENGTH
  RECORD_BYTES             = 2880
  FILE_RECORDS             = 12
  ^FITS_HEADER             = ("0101_SHF.DAT",     1 <BYTES>)
  ^HEADER_TABLE            = ("0101_SHF.DAT", 25921 <BYTES>)

  OBJECT                   = FITS_HEADER
    HEADER_TYPE            = FITS
    INTERCHANGE_FORMAT     = ASCII
    RECORDS                = 7
    BYTES                  = 20160
    ^DESCRIPTION           = "FITS.TXT"
  END_OBJECT               = FITS_HEADER

  OBJECT                   = HEADER_TABLE
    NAME                   = HEADER_PACKET
    INTERCHANGE_FORMAT     = BINARY
    ROWS                   = 965
    COLUMNS                = 1

    ROW_BYTES              = 2
    DESCRIPTION            = "This is the HST standard header packet
                           containing observation parameters.  It is
                           stored as a sequence of 965 two-byte
                           integers.  For more detailed information,
                           contact Space Telescope Science Institute."

    OBJECT                 = COLUMN
      NAME                 = PACKET_VALUES
      DATA_TYPE            = MSB_INTEGER
```

```
      START_BYTE          = 1
      BYTES               = 2
    END_OBJECT            = COLUMN
  END_OBJECT              = HEADER_TABLE

END_OBJECT                = UNCOMPRESSED_FILE
END
```

I.6.4 ZIPINFO.TXT Example

While the ZIPINFO.TXT file is not required, it is strongly recommended that this file be included as part of the process of documenting the contents of a zip file. The following is an example ZIPINFO.TXT file and the type of information that should be included in the ZIPINFO.TXT file:

```
PDS_VERSION_ID            = PDS3
RECORD_TYPE               = STREAM

OBJECT                    = TEXT
  PUBLICATION_DATE        = 1999-07-26
  NOTE                    = "This file provides an overview of the ZIP
                            file format."

END_OBJECT                = TEXT
END

Many of the files in this data set are compressed using Zip format.
They are all indicated by the extension ".ZIP".  ZIP is a utility that
compresses files and also allows for multiple files to be stored in a
single Zip archive.  You will need the UNZIP utility to extract the
files.

The SOFTWARE directory on this volume contains a complete description
of the Zip file format and also the complete source code for the UNZIP
utility.  The file format and file decompression algorithms are
described in the file SOFTWARE/APPNOTE.TXT.

It is far simpler to obtain a pre-built binary of the UNZIP application
for your platform.  Binaries for most platforms are available from the
Info-ZIP web site, currently at this URL:

            http://www.info-zip.org/

The same information can also be found a the PDS Engineering Node's web
site, currently at:

            http://pds.jpl.nasa.gov/
```

(This page intentionally left blank.)